



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
[www.uspto.gov](http://www.uspto.gov)

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/933,493	08/20/2001	Stefan M. van den Oord	MOBJ-01000US0	6164
23910 7590 03/31/2010 FLIESLER MEYER LLP 650 CALIFORNIA STREET 14TH FLOOR SAN FRANCISCO, CA 94108				
EXAMINER				
BURGEISS, BARBARA N				
ART UNIT		PAPER NUMBER		
2457				
NOTIFICATION DATE		DELIVERY MODE		
03/31/2010		ELECTRONIC		

**Please find below and/or attached an Office communication concerning this application or proceeding.**

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

OFFICEACTIONS@FDML.COM

**Office Action Summary****Application No.**

09/933,493

**Applicant(s)**

VAN DEN OORD ET AL.

**Examiner**

BARBARA N. BURGESS

**Art Unit**

2457

**-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --**  
**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

**Status**

- 1) ☒ Responsive to communication(s) filed on 19 February 2010.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

**Disposition of Claims**

- 4) ☒ Claim(s) 1-15, 17, 24-27, 29-60, 63 and 64 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.
- 6) ☒ Claim(s) 1-15, 17, 24-27, 29-60, 63 and 64 is/are rejected.
- 7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

**Application Papers**

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on \_\_\_\_\_ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.  
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

**Priority under 35 U.S.C. § 119**

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some \* c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
  2. ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
  3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

\* See the attached detailed Office action for a list of the certified copies not received.

**Attachment(s)**

- 1) ☒ Notice of References Cited (PTO-892)
- 2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) ☒ Information Disclosure Statement(s) (PTO-85/86)  
Paper No(s)/Mail Date 2-19-10
- 4) ☐ Interview Summary (PTO-413)  
Paper No(s)/Mail Date \_\_\_\_\_
- 5) ☐ Notice of Informal Patent Application
- 6) ☐ Other: \_\_\_\_\_

### **DETAILED ACTION**

This Office Action is in response to Request for Continuation Examination (RCE) filed February 19, 2010. Claims 1-15, 17, 24-27, 29-60, 63-64 are presented for further examination.

### ***Claim Rejections - 35 USC § 103***

1. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

2. Claims 1- 15, 17, 29-51-60 are rejected under 35 U.S.C. 103(a) as being unpatentable over Yano et al. (hereinafter "Yano", US Patent Publication 2006/0184546 A1) in view of Andrew Clinick (hereinafter "Clinick", Remote Scripting).

As per claims 1, 29, 32-33, Yano discloses a system, user interface mechanism, and method of providing session-based retrieval and at a client system of string-based content from a server comprising:

- A communication protocol that enables connection over a network between a client system and a server system, and allows the client system to send via the network, and within a single session between the client system and the server system, a lengthening string composed of a plurality of consecutively input

characters, to query the server system for string-based content (paragraphs [0032-0033, 0039, 0042]);

- A client object, in communication with a client software at the client system and with the communication protocol, wherein the client object receives, as input, consecutive additional characters from the client software, and while each of the consecutive additional characters are being received as input, transmits via the network to a server object at the server system one or more corresponding consecutive queries, within the same session, to retrieve content from the server system, wherein each of the consecutive queries lengthens the query string by additional characters, to form a lengthening string for retrieving matching content from the server system (paragraphs [0053, 0077, 0093, 0117]);
- A server object, in communication with the server system, and with the client object via the communication protocol, wherein the server object in response to receiving the consecutive queries that form the lengthening string, automatically uses the lengthening string to query and retrieve content information from the server system that matches the lengthening string, and wherein the server object returns, while the additional characters are being input and the string is being lengthened during the session, increasingly matching content information to the client object for immediate use by the client system (paragraphs [0035, 0045, 0061]).

Yano does not explicitly disclose:

- The session between the client system and the server system;

- A communication protocol that enables an asynchronous session based connection between a client system and a server system;
- While the consecutive queries are being transmitted and the lengthening query string is being modified, asynchronously returns content information which increasingly matches the modified lengthening string, to the client object for immediate use by the client system;
- Asynchronously receiving consecutive responses from the server as the characters are being input.

However, in an analogous art, Clinick teaches an applet running on a client device communicating to the server. Every script call results in a request being sent via the applet to the server. The result of the query can be returned immediately known as an asynchronous call. The user can continue to type characters while Remote Scripting awaits results to be returned (pages 2, 4).

Therefore, one of ordinary skill in the art at the time the invention was made would have found it obvious to incorporate or implement Clinick's asynchronously returning responses in Yano's system giving the ability to return information in a more intuitive manner.

As per claim 2, Yano discloses the system of claim 1 wherein said client object operates on or at a first computer and said server object operates on or at a second computer and wherein both of said first and said second computers are connected via a network protocol (paragraphs [0033-0035]).

As per claim 3, Yano discloses the system of claim 1 wherein said server object and said client object runs on the same computer (paragraph [0037]).

As per claim 4, Yano discloses the system of claim 1 wherein said server object runs on a plurality of separate computers, and wherein said client queries received during the session are distributed over said separate computers (paragraph [0070]).

As per claim 5, Yano discloses the system of claim 1 wherein said server object stores previously received strings and returns said stored strings to the client in response to new client queries received during the session, without accessing said content (paragraph [0042, 0058]).

As per claim 6, Yano further discloses the system of claim 1 wherein said client software is embedded into a software application that provides a visual interface that indicates to an operator that the server object is currently using the lengthening query string against the content of the server system to query and retrieve content information from the server system and allows the operator to add additional characters to lengthen the query string, while simultaneously receiving and displaying increasingly matching results in consecutive responses from the server (paragraph [0053, 0065, 0077]).

As per claim 7, Yano discloses the system of claim 1 wherein said client software is used as a content engine for another software system (paragraph [0057]).

As per claim 8, Yano discloses the system of claim 1 wherein said client software accumulates a plurality of said single character queries as they are entered into the client, before sending them together as a single query string to said server (paragraph [0057]).

As per claim 9, Yano discloses the system of claim 1 wherein said client software stores previously received responses from the server in a cache at the client and uses the previously received responses as the response to a new query by the user, without reassessing the server (paragraph [0065]).

As per claim 10, Yano discloses the system of claim 1 wherein said client software stores a pre-defined string and automatically transmits it to the server as the client software is first accessed, and wherein additional entry of query characters is not required before server responses are sent to the client (paragraph [0075, 0093]).

As per claim 11, Yano discloses the system of claim 1 wherein said server stores the state of query and response of the client software, and restores the state of the client software after any interruption in said communication protocol (paragraph [0084]).

As per claim 12, Yano discloses the system of claim 1 where said client software adds a qualifier to the query that is passed to the server, whereby the server can use said qualifier to execute the query and return appropriate results based on both the query string and its qualifier (paragraph [0069]).

As per claim 13, Yano discloses the system of claim 1 where said client software identifies a user of the system to the server whereby the server can store statistics and provides a history of queries and corresponding responses appropriate to said user (paragraph [0125, 0141]).

As per claim 14, Yano discloses the system of claim 1 where said server system comprises a server tier and a syndication tier, and wherein said client software communicates to the server tier on a single computer, and wherein each query is forwarded by the server tier and the syndication tier to an appropriate syndicate of content channels connected to the server tier on a different computer (paragraph [0137]).

As per claim 15, Yano discloses the system of claim 1 where said server applies a content dependent pattern and filter to characters received from the client before queries are matched against the content (paragraph [0050, 0065]).



As per claim 17, Yano further discloses the system of claim 1 where server responses comprise lists of strings, wherein each string is accompanied by corresponding metadata, whereby the metadata contains logical links to other data sources of Uniform Resource Identifiers (paragraph [0088]).

As per claim 30, Yano discloses the method of claim 29 wherein the server object matches each query received from the client against an in-memory cache, and returns cached content to the client without accessing said content engine, unless the cached content has expired since it was last received from said content engine (paragraph [0105]).

As per claim 31, Yano discloses the method of claim 29, wherein the server analyzes the time between said consecutive queries received from each client system, and skips selected ones of said consecutive queries to reduce network communications and the load on said content engine (paragraph [0075]).

As per claim 34, Yano discloses the system of claim 1, whereby the client object indicates the selection of the content sources to be queried to the server when said session is initiated and when content source selection changes are needed thereafter, without needing to embed said content source selection with each of said consecutive string-based queries (paragraph [0095]).

As per claim 35, Yano discloses the system of claim 1 .whereby said session is shared by multiple client objects that exchange messages with the same server system, whereby each client object identifies a different content source selection to which said consecutive queries from the individual client object will be mapped by its corresponding server object (paragraph [0135]).

As per claims 36-37, Yano discloses a system and method for providing session-based searching of string-based content from a server, comprising:

- a user interface at a plurality of clients that allows a user at each of the plurality of clients to enter input as a string of consecutively input queries to query the server for string-based content, wherein each consecutive query lengthens the query string by one or more consecutive additional characters (paragraphs [0053, 0077, 0093, 0117]);
- a communication protocol that transmits over a network, via a client object at each said client, to a server object at the server, the plurality of consecutive queries, to retrieve content from the server, wherein each consecutive additional character is immediately transmitted to the server object while the user is entering the additional characters in the user interface, to form an lengthening query string for retrieving content from the server (paragraphs [0032-0033, 0039, 0042]);
- a server object which in response to receiving each query as it is being lengthened or shortened by the one or more additional characters, automatically matches the lengthening query string against the content of the server, and, as

the user of a particular client is entering queries, modifies the user interface by returning increasingly relevant server content information to the client object for immediate display to the user (paragraphs [0035, 0045, 0061]).

Yano does not explicitly disclose:

- A communication protocol that enables an asynchronous session based connection between a client system and a server system;
- While consecutive queries are being transmitted and the lengthening query string is being modified, asynchronously modifies the user interface by returning server content information which increasingly matches the modified lengthening string, to the client object for immediate use by the client system.

However, in an analogous art, Clinick teaches an applet running on a client device communicating to the server. Every script call results in a request being sent via the applet to the server. The result of the query can be returned immediately known as an asynchronous call. The user can continue to type characters while Remote Scripting awaits results to be returned (pages 2, 4).

Therefore, one of ordinary skill in the art at the time the invention was made would have found it obvious to incorporate or implement Clinick's asynchronously returning responses in Yano's system giving the ability to return information in a more intuitive manner.

As per claim 38, Yano discloses the system of claim 1 wherein the client software is used to one of display suggestions, perform auto-completion, or provide type-ahead

functionality, based on matching string-based data queried in a database by the server object on the server system (paragraph [0058]).

As per claim 39, Yano discloses the system of claim 1 wherein the client software one of validates or checks the input string based on responses received from the server object on the server system (paragraph [0065-00696]).

As per claim 40, Yano discloses the system of claim 1 wherein the lengthening query string is one of a part of a name, email address, URL, phone number, or other typed string that can be normalized as a simple term, definitional term, relational term, quote, simple number, compound number, date, URL, e-mail address, phone number, or XML formatted data corresponding to a DTD or schema (paragraph [0076, 0095]).

As per claim 41, Yano discloses the system of claim 1 wherein the matching content returned by the server object contains one of a term from a thesaurus system, result received from a search and retrieval system, text from a reference work, match from an address book, appropriate instructions or actions to be taken received from a control system, entry from a dictionary, thesaurus, or encyclopedia, match from a commercial products database, quote from a literary quotes library, real-time stock quote, content from a real-time news service, Internet advertisement, result of a complex function,

translation received from a language translation engine, entry from a classification scheme, match from a lookup list such as cities or countries in an order form, match from a auto-complete history or a language code, creation date, modification date, pronunciation, meaning, possible use, synonym, reference, scope note, notation, source, UDC coding, description, product code, category, price, currency, stock symbol, company name, stock quote, machine instruction or a city or a country ([0069, 0125]).

As per claim 42, Yano discloses the system of claim 1 wherein the server object retrieves the matching string-based data from an in-memory cache of responses to previous queries (paragraph [0068, 0076]).

As per claim 43, Yano discloses the system of claim 1 wherein the server object, in communication with a content access module object, retrieves matching content from multiple content engines, and wherein the server object embeds the query string into a native query for each type of content engine (paragraph [0069, 0081]).

As per claim 44, Yano discloses the system of claim 43 wherein the content engine is a SQL database or a search engine (paragraph [0057, 0090]).

As per claim 45, Yano discloses the system of claim 1 wherein the client software displays arrow symbols to indicate the availability or lack of matching results (paragraph [0076]).

Art Unit: 2457

As per claim 46, Yano discloses the system of claim 1 wherein the client software displays a checkmark symbol if only one match was found for the query string (paragraph [0118]).

As per claim 47, Yano discloses the system of claim 1 wherein a plurality of client objects are logically linked to multiple content sources on the server system, so that results received and returned by each corresponding server object are the result of a match to both the lengthening query string and values contained by one or more of the other client objects (paragraph [0077, 0093]).

As per claim 48, Yano discloses the system of claim 1 wherein the client software displays images and/or movies corresponding to individual matches received from the server system (paragraph [0069]).

As per claim 49, Yano discloses the system of claim 1 wherein only a specific requested or expected range of matches are returned to the client object any one time (paragraph [0136]).

As per claim 50, Yano discloses the system of claim 1 wherein the client software runs in a web browser (paragraph [0057]).

As per claim 51, Yano discloses the system of claim 1 wherein the client software displays a symbol inside of an input field to indicate the presence and availability of said system to text entered into said input field (paragraph [0065]).

As per claims 52-53, Yano discloses a system and method for suggesting data as a response to client requests, comprising:

a server configured to receive requests from a plurality of clients for content (paragraphs [0035, 0045, 0061]);

an interface to a plurality of databases or data sources of content information coupled to said server (paragraphs [0037, 0042, 0045, 0058]);

a communication protocol that provides a session connection between a client and the server, and allows the client to send, as part of the same session, a plurality of queries to query the server for content, wherein each of the plurality of queries are consecutive and form an increasingly focused query string for retrieving content from the server, and wherein each subsequent one of the plurality of queries extends the query string by one or more additional characters (paragraphs [0032-0033, 0039, 0042]); and

wherein said server simultaneously applies the increasingly focused query string against the plurality of databases or data sources as it is being extended, and suggests a set of increasingly appropriate content or search criteria from the plurality of databases, to the client, for further use by the client within the same session (paragraphs [0035, 0045, 0061]).

Yano does not explicitly disclose:

Provide a user interface for input queries;

Extending the query string in the user interface by one or more additional characters;

Receive each subsequent one of the plurality of queries while it is being entered into the user interface;

While the user is entering the one or more characters, suggests a set of increasingly appropriate content.

However, in an analogous art, Clinick teaches an applet running on a client device communicating to the server. Every script call results in a request being sent via the applet to the server. The result of the query can be returned immediately known as an asynchronous call. The user can continue to type characters while Remote Scripting awaits results to be returned (pages 2, 4).

Therefore, one of ordinary skill in the art at the time the invention was made would have found it obvious to incorporate or implement Clinick's providing a user interface and the user entering one or more characters in Yano's system giving the ability to return information in a more intuitive manner.

As per claim 54, Yano discloses a system comprising:



a client object on a client computer and a server object on a server computer, whereby the client computer and the server computer are linked by a network so that they can exchange information (0033-0036));

wherein the client object is linked to an input element in a user interface that allows a user to enter textual information comprising characters and strings to create incremental user input comprising a mutating string of characters (paragraphs [0042, 0065]);

wherein said user input is transmitted by the client object to the server object as one or more consecutive additional characters while said user input is being formed by a specific user during a user session (paragraph [0045, 0048]);

wherein the server object uses said user input received from the client object to query data from one or more content sources, and to return result strings matching said user input from said server computer while the input is being formed on the client computer; and wherein the client object displays said results in a display element in the user interface on the client computer (paragraphs [0035, 0045, 0061]).

Yano does not explicitly disclose:

- A communication protocol that enables an asynchronous session based connection between a client system and a server system;
- While the consecutive additional characters are being transmitted and the mutating query string is being modified, returns result strings matching said user input asynchronously from said server computer.

However, in an analogous art, Clinick teaches an applet running on a client device communicating to the server. Every script call results in a request being sent via the applet to the server. The result of the query can be returned immediately known as an asynchronous call. The user can continue to type characters while Remote Scripting awaits results to be returned (pages 2, 4).

Therefore, one of ordinary skill in the art at the time the invention was made would have found it obvious to incorporate or implement Clinick's asynchronously returning responses in Yano's system giving the ability to return information in a more intuitive manner.

As per claim 55, Yano discloses the system of claim 54 whereby said client object is embedded in an object that is part of a web page and appears in a web browser on the client computer (paragraph [0041, 0050]).

As per claim 56, Yano discloses the system of claim 54 whereby each of said matching result strings is accompanied by a key that identifies each result as it was retrieved from the one or more content sources, whereby the key of selected results can be used for sorting and merging and sorting in the server computer and are transmitted back to the client object for use on the client system (paragraph [0060, 0065]).

As per claim 57, Yano discloses the system of claim 54 whereby the client object accumulates the user input for an amount of time before sending the resulting string of characters to the server object as a single consolidated query string, to decrease network traffic and decrease the load on the server computer (paragraph [0042, 0048]).

As per claim 58, Yano discloses the system of claim 57 whereby the input element on the client computer contains a visual object displayed within the display element that indicates to the user that user input was sent to the server object whereby the visual object keeps changing while matching results are being awaited from the server system, and whereby said visual object first changes when the user enters textual information, and before the user input is sent to the server, indicating to the user that the user input is being accumulated by the client object before sending it to the server object (paragraph [0152]).

As per claim 59, Yano discloses the system of claim 54 whereby the server object provides one or more content channels for retrieving configurable sets of data available on the server computer, whereby each content channel defines a logical data set to be retrieved from the one or more data sources (paragraph [0090, 0131]).

Art Unit: 2457

As per claim 60, Yano discloses the system of claim 54 whereby the server object caches the result data received from said content sources and uses said cached result data as a response to later client requests originating from the same client object or from a different client object (paragraph [0102, 0127]).

3. Claims 24-27, 63-64 are rejected under 35 U.S.C. 103(a) as being unpatentable over Yano et al. (hereinafter "Yano", US Patent Publication 2006/0184546 A1) in view of Andrew Clinick (hereinafter "Clinick", Remote Scripting) and in further view of Vora et. al (hereinafter "Vora", US Patent 6,539,379 B1).

As per claims 24, 63-64, Yano discloses a user interface mechanism and system comprising:

- a communication protocol that enables connection over a network between a client system and a server system, and allows the client system to send via the network, and within a single session between the client system and the server system, a lengthening string composed of a plurality of consecutively input characters, to query the server system for string-based content (paragraphs [0032-0033, 0039, 0042]);
- A client object, in communication with a client software at the client system and with the communication protocol, wherein the client object receives additional characters from the client software, and as consecutive characters are being received, transmits via the network to a server object at the server system a plurality of consecutive queries, within the same session, to retrieve content from the server

system, wherein each consecutive lengthens the query string by additional characters, to form a lengthening string for retrieving matching content from the server system (paragraphs [0053, 0077, 0093, 0117]);

- A server object, in communication with the server system, and with the client object via the communication protocol, wherein the server object in response to receiving the consecutive queries that form the lengthening string, automatically uses the lengthening string to query and retrieve content information from the server system that matches the lengthening string, and wherein the server object returns, while the additional characters are being input and the string is being lengthened during the session, increasingly matching content information to the client object for immediate use by the client system (paragraphs [0035, 0045, 0061]).

Yano does not explicitly disclose:

- The session between the client system and the server system.
- A communication protocol that enables an asynchronous session based connection between a client system and a server system;
- While the consecutive queries are being transmitted and the lengthening query string is being modified, asynchronously returns content which increasingly matches the modified lengthening string to the client object for immediate use by the client system;
- Asynchronously receiving consecutive responses from the server as the characters are being input.

However, in an analogous art, Clinick teaches an applet running on a client device communicating to the server. Every script call results in a request being sent via the applet to the server. The result of the query can be returned immediately known as an asynchronous call. The user can continue to type characters while Remote Scripting awaits results to be returned (pages 2, 4).

Therefore, one of ordinary skill in the art at the time the invention was made would have found it obvious to incorporate or implement Clinick's asynchronously returning responses in Yano's system giving the ability to return information in a more intuitive manner.

Yano, in view of Clinick, does not explicitly disclose:

a session connection indicator, said session connection indicator displayed within a first portion of the input field, for indicating the availability of a connection between said client application and said content server; and

a status indicator, said status indicator displayed within the first or a second portion of the input field, for indicating during said session both the status of increasingly available content at said content server for selection by said user at that input field, and that the server object is currently using the lengthening query string against the content of the server system to query and retrieve content information from the server system.

However, in an analogous art, Vora teaches indicating whether the requests have been completed or denied (column 1, lines 35-50, column 2, lines 19-30).

Therefore, one of ordinary skill in the art at the time the invention was made would have found it obvious to incorporate or implement Vora's status indicator and session connection indicator in Yano's system in order to determine the completion of a request.

As per claim 25, Yano discloses the mechanism of claim 24, wherein said user interface element is an application input field (paragraph [0073]).

As per claim 26, Yano does not explicitly disclose the mechanism of claim 24, wherein said session indicator displays a triangular display element to indicate the presence of said session, and does not display said triangular display element to indicate the absence of said session.

However, in an analogous art, Vora teaches indicating whether the requests have been completed or denied (column 1, lines 35-50, column 2, lines 19-30).

Therefore, one of ordinary skill in the art at the time the invention was made would have found it obvious to incorporate or implement Vora's status indicator and session connection indicator in Yano's system in order to determine the completion of a request.

As per claim 27, Yano, in view of Clinick, does not explicitly disclose the mechanism of claim 24, wherein said status indicator displays one, or a plurality of arrow display

elements to indicate the transfer of data from said client application to said server during said session, and the presence of available session-specific content at said server.

However, in an analogous art, Vora teaches indicating whether the requests have been completed or denied (column 1, lines 35-50, column 2, lines 19-30).

Therefore, one of ordinary skill in the art at the time the invention was made would have found it obvious to incorporate or implement Vora's status indicator and session connection indicator in Yano's system in order to determine the completion of a request.

### ***Response to Arguments***

#### **The Office notes the following argument(s):**

- (a) Yano does not appear to describe a keyword being sent from the browser to the search engine as a plurality of consecutive queries, or even as a plurality of queries.
- (b) None of the cited references appears to disclose a communication protocol that enables an asynchronous connection over a network between a client system and server system.

#### **In response to:**

- (a) Applicant's arguments filed have been fully considered but they are not persuasive.



Yano teaches a user entering each character of the keyword into the search engine. The system performs a search for the keyword as each character is entered (paragraphs [0079-0083]).

Therefore, Yano discloses a plurality of consecutive queries.

(b) Applicant's arguments have been considered but are moot in view of the new ground(s) of rejection.

### ***Conclusion***

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Barbara N Burgess whose telephone number is (703) 305-3366. The examiner can normally be reached on M-F (8:00am-4:00pm).

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Ario Etienne can be reached on (703) 308-7562. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306. Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

/Barbara N Burgess/

Application/Control Number: 09/933,493

Page 25

Art Unit: 2457

Examiner, Art Unit 2457

March 26, 2010